Please amend the claims as follows:

Claims 1-14 (canceled)

15. (Currently Amended) A computer implemented replacement selection method comprising:

creating a tree with a root node, multiple non-root nodes, and leaf nodes, each node having a first integer identifying one of at least three input data streams and a status identifier identifying a status of each of said input data streams, wherein a number of input streams is odd;

processing a data item from each of said at least three input streams by placing one data item from each of said three input streams in each of said leaf nodes of said tree, respectively;

promoting said data items of said leaf nodes through said tree from a leaf node to said root node, including comparing a first status identifier of a first node with a second status identifier of a second node, promoting one of said data items of one said nodes to a node in a next level in said tree responsive to a status identifier value, and switching a next comparison to a third status identifier of a third node with one of said other nodes in said tree;

merging data items from said input streams into a single output stream from said promoted data items, said output stream being a stream separate from said input streams, by copying compared data items promoted to said root node of said tree, wherein the step of merging said data items into a single output stream includes resolving comparison of duplicate identifiers comprising:

comparing said status identifier of two of said nodes;

omitting a second node identifier comparison when a first data item in a first of said nodes remained in said node and was a duplicate of a second data item in a second of said nodes, wherein said second data item was promoted to a next ~~hierarchical node~~ level in said tree; and

2

promoting said data item from said first node to ~~a~~ <u>said</u> next <u>level</u> ~~hierarchical node~~ in said tree absent a comparison of said data items in said nodes when a next time ~~a~~ said first node is designated for a key comparison; and

said single output stream comprised of said merged data items.

16. (Currently Amended) The method of claim 15, wherein the step of promoting said data item from said first node to a next ~~hierarchical node~~ <u>level</u> in said tree absent a comparison of said data items includes skipping a comparison of said remembered first data item with data items in other nodes of said tree.

17. (Original) The method of claim 15, wherein said status identifier has a value corresponding to empty, duplicate, merging, and done.

18. (Previously Presented) The method of claim 17, wherein said status identifier is an integer variable, and said status identifier value corresponding to empty is a value of zero, said status identifier value corresponding to duplicate is a value of one, said status identifier value corresponding to merging is a value of two, and a status identifier value corresponding to done is a value of three.

19. (Original) The method of claim 15, wherein the replacement selection method is a loser-oriented selection tree,

20. (Currently Amended) The method of claim 15, wherein the step of promoting said data item from said first node to a next ~~hierarchical node~~ <u>level</u> in said tree absent a comparison of said data items includes avoiding exhausting promoting duplicate data items from a single input data stream.

21. (Currently Amended)   A system for processing computer-readable data comprising:
    a tree with a root node, multiple non-root nodes, and leaf nodes, each node having

3

a first integer to identify one of at least three input data streams and a status identifier to identify a status of each of said input data streams;

a data item to be processed from each of said at least three input streams by placement of one data item from each of said three input stream in each of said leaf nodes of said tree, respectively;

instructions to promote said data items of said leaf nodes through said tree from a leaf node to said root node, including comparison of a first status identifier of a first node with a second status identifier of a second node, to promote one of said data items of one said nodes to a node in another level of said tree responsive to a status identifier value, and to switch a next comparison to a third status identifier of a third node with one of said other nodes in said tree;

a merge of said data items from said input streams into a single output stream from said promoted data items, said output stream being separate from said input streams wherein merge of said input streams into said single output stream includes~~;~~ instructions to resolve comparison of duplicate identifiers comprising:

comparison of a status identifier of two of said nodes;

omission a second node identifier comparison when a first data item in a first of said nodes remained in said node and was a duplicate of a second data item in a second of said nodes, wherein said second data item was promoted to a next level ~~hierarchical node~~ in said tree; and

promotion of said data item from said first node to ~~a next hierarchical node~~ said next level in said tree absent a comparison of said data items in said nodes when a next time ~~a~~ said first node is designated for a key comparison; and said single output stream comprised of said merged data items.

22. (Currently Amended) The system of claim 21, wherein the instructions to promote said data item from said first node to a next ~~hierarchical node~~ level in said tree absent a comparison of said data items includes skip of a comparison of said first data item with data items in other nodes of said tree.

23. (Original) The system of claim 21, wherein said status identifier has a value corresponding to empty, duplicate, merging, and done.

24. (Previously Presented)   The system of claim 23, wherein said status identifier is an integer variable, and said status identifier value corresponding to empty is a value of zero, said status identifier value corresponding to duplicate is a value of one, said status identifier value corresponding to merging is a value of two, and a status identifier value corresponding to done is a value of three.

25. (Original) The system of claim 21, wherein the promotion instruction is a loser-oriented selection tree.

26. (Currently Amended) The system of claim 21, wherein the instructions to promote said data item from said first node to a next ~~hierarchical node~~ <u>level</u> in said tree absent a comparison of said data items includes instructions to avoid exhausting promotion of duplicate data items from a single input data stream.

27. (Currently Amended)   An article comprising:

a computer-readable data storage medium;

instructions in the medium for implementing a replacement selection method comprising:

instructions for creating a tree with a root node, multiple non-root nodes, and leaf nodes, each node having a first integer identifying one of at least three input data streams and a status identifier identifying a status of each of said input data streams;

instructions for processing a data item from each of said at least three input streams by placing one data item from each of said three input streams in each of said leaf nodes of said tree, respectively;

instructions for promoting said data items of said leaf nodes through said tree from a leaf node to said root node, including comparing a first status identifier of a first

5

node with a second status identifier of a second node, promoting one of said data items of one said nodes to a node in a next level in said tree in response to a status identifier value, and switching a next comparison to a third status identifier of a third node with one of said other nodes in said tree;

instructions for merging data items from said input streams into a single output stream from said promoted data items, said output stream being a stream separate from said input streams, by copying compared data items promoted to said root node of said tree, wherein the instructions for merging said data items into a single output stream includes instructions for resolving comparison of duplicate identifiers comprising:

comparing said status identifier of two of said nodes;

omitting a second node identifier comparison at each node when a first data item in a first of said nodes remained in said node and was a duplicate of a second data item in a second of said nodes, wherein said second data item was promoted to a next ~~hierarchical node~~ level in said tree; and

promoting said data item from said first node to said ~~a~~ next ~~hierarchical node~~ level in said tree absent a comparison of said data items in said nodes when a next time ~~a~~ said first node is designated for a key comparison; and

said single output stream comprised of said merged data items.

28. (Currently Amended) The article of claim 27, wherein the instructions for promoting said data item from said first node to said ~~a~~ next ~~hierarchical node~~ level in said tree absent a comparison of said data items includes skipping a comparison of said first data item with data items in other nodes of said tree.

29. (Original) The article of claim 27, wherein said status identifier has a value corresponding to empty, duplicate, merging, and done.

30. (Previously Presented) The article of claim 29, wherein said status identifier is an integer variable, and said status identifier value corresponding to empty is a value of zero, said

6

status identifier value corresponding to duplicate is a value of one, said status identifier value corresponding to merging is a value of two, and a status identifier value corresponding to done is a value of three.

31. (Original) The article of claim 27, wherein the replacement selection method is a loser-oriented selection tree,

32. (Currently Amended) The article of claim 27, wherein the instructions for promoting said data item from said first node to a next ~~hierarchical node~~ level in said tree absent a comparison of said data items includes avoiding exhausting promoting duplicate data items from a single input data stream.